

A Low-Complexity Joint Synchronization and Detection Algorithm for Single-Band DS-CDMA UWB Communications

Lars P. B. Christensen

Information and Mathematical Modelling, Technical University of Denmark, 2800 Kongens Lyngby, Denmark
Email: lc@imm.dtu.dk

Received 1 October 2003; Revised 2 June 2004

The problem of asynchronous direct-sequence code-division multiple-access (DS-CDMA) detection over the ultra-wideband (UWB) multipath channel is considered. A joint synchronization, channel-estimation, and multiuser detection scheme based on the adaptive linear minimum mean square error (LMMSE) receiver is presented and evaluated. Further, a novel nonrecursive least-squares algorithm capable of reducing the complexity of the adaptation in the receiver while preserving the advantages of the recursive least-squares (RLS) algorithm is presented.

Keywords and phrases: ultra-wideband, direct-sequence code-division multiple-access, multiuser detection, low-complexity adaptive receivers, synchronization.

1. INTRODUCTION

Over the last couple of years, the interest in ultra-wideband (UWB) wireless communications has been growing. Among the reasons for this increased awareness of UWB are the promises of low-power, high-bitrate wireless connections without the need for spectrum allocation, and the approval of the technology by authorities as, for example, the American FCC [1].

UWB signals for wireless communication typically have a bandwidth of several GHz and can be utilized in many ways each presenting the designer with tradeoffs between cost, power, bitrate, range, and the number of users supported. The system considered in this paper is a single-band UWB direct-sequence code-division multiple-access (DS-CDMA) receiver with all signal processing done on the received signal sampled directly from an amplified and filtered antenna signal. This enables the removal of traditional up- and down-converters present in today's narrowband transceivers at the expense of increasing the required sampling rate and thus the complexity of the signal processing. It is therefore of great interest to reduce the complexity of such receivers to make them feasible.

The receiver considered is fully adaptive making it possible to track changes not only in the multipath channel, but also in the received pulse shape. This is desirable in order to maximize performance even under conditions distorting the received pulse shape as discussed in [2], but distortions orig-

inating from the electromagnetic propagation environment can also be adaptively compensated for.

Combined LMMSE synchronization and detection for DS-CDMA systems have already been studied (see, e.g., [3, 4, 5, 6, 7]). This paper is a continuation of [8] extended with the synchronization method in [3], but having a low-complexity adaptive algorithm with recursive least-squares (RLS)-speed convergence. Furthermore, this paper uses the channel model presented in [9] instead of the model in [8] as the latter may prove too optimistic for typical office use as a result of the larger dimensions typically present in office environments.

The rest of this paper is organized as follows. Section 2 describes the system model used throughout this paper. In Section 3, the LMMSE receiver is presented as a benchmark of how well the adaptive receiver outlined by Section 4 performs compared to the best possible linear receiver. Synchronization of the receiver is covered in Section 5 and Section 6 presents simulations of the receiver. Section 7 concludes the paper with final remarks.

2. SYSTEM MODEL

The receiver considered is the adaptive LMMSE receiver with the system model being capable of supporting K asynchronous users each operating in their respective multipath radio channel. The desired user is, without loss of generality, assumed to be user 1.

2.1. Transmitted signal

The pulse shape used for transmission $p(t)$ is of duration T_{mono} and is assumed normalized to the unit energy. This pulse shape is traditionally called a monocycle in UWB terms and it is typically modeled as the q th derivative of a Gaussian pulse [10], which is also the case in this paper. This makes it possible to include the differentiation performed by the antennas and further control the spectrum of the transmitted signal. To include the effect of asynchronous operation between users, the delay $\tau^{(k)}$ is introduced for the k th user.

Next, the binary DS spreading code $c^{(k)}(i) \in \{-1, +1\}$, for $i = 1, \dots, N_c$, is used to separate the different users and provide a processing gain of N_c , where N_c indicates the number of coded monocycles transmitted for each bit of information. Finally, the binary information given by $b^{(k)}(j) \in \{-1, +1\}$ is assumed to be a memoryless random source with equal probability of $+1$ and -1 . The modulation considered is binary phase shift keying (BPSK) and the transmitted signal from the k th user can therefore be written as

$$\begin{aligned} s^{(k)}(t) &= \sum_{j=-\infty}^{\infty} b^{(k)}(j) \varphi^{(k)}(t - jT_b - \tau^{(k)}) \\ &= \sum_{j=-\infty}^{\infty} b^{(k)}(j) \sum_{i=0}^{N_c-1} c^{(k)}(i) p(t - jT_b - iT_{\text{mono}} - \tau^{(k)}). \end{aligned} \quad (1)$$

The waveform $\varphi^{(k)}(t)$ has duration $T_b = N_c T_{\text{mono}}$ holding N_c monocycles coded by the user's spreading code.

2.2. Radio channel

To include the effects of a realistic multipath environment, the radio channel model given in [9] is used. The impulse response of this model for the k th user can be written as

$$h^{(k)}(t) = \sum_{l=0}^{L-1} a_l^{(k)} \delta(t - lT_{\text{ch}}), \quad (2)$$

where T_{ch} is the temporal spacing between the L multipath components and $\delta(t)$ is the Dirac delta function. The amplitude of the l th multipath component is given by $a_l^{(k)}$ and is assumed to be constant over time. Convolution of the transmitted signal of the k th user given by (1) with its respective impulse response given by (2), the contribution from this user onto the received signal can be written as

$$r^{(k)}(t) = \sum_{l=0}^{L-1} a_l^{(k)} s^{(k)}(t - lT_{\text{ch}}) \quad (3)$$

and the received signal is therefore

$$\begin{aligned} r(t) &= \sum_{k=1}^K r^{(k)}(t) + n(t) \\ &= \sum_{k=1}^K \sum_{l=0}^{L-1} a_l^{(k)} s^{(k)}(t - lT_{\text{ch}}) + n(t) \end{aligned} \quad (4)$$

with $n(t)$ being white Gaussian noise with zero mean and variance σ^2 leading to the signal-to-noise ratio (SNR) at the receiver being defined as

$$\text{SNR} = \frac{\sum_{l=0}^{L-1} |a_l^{(1)}|^2}{\sigma^2}. \quad (5)$$

3. THE LMMSE RECEIVER

In the receiver an antialiasing filter processes the received signal before it is uniformly sampled and fed directly into a tapped-delay-line filter with the input given by the vector

$$\mathbf{r}(j) = [r(jT_b), r(jT_b + T_s), \dots, r(jT_b + (N-1)T_s)]^T, \quad (6)$$

where N is the length of the tapped-delay-line filter with a sample spacing of T_s . In order to be able to capture the entire multipath energy spread out by the channel model, the number of filter taps must be at least

$$N_{\text{full}} = \left\lceil \frac{T_b + (L-1)T_{\text{ch}}}{T_s} \right\rceil \quad (7)$$

with the operator $\lceil x \rceil$ returning the smallest integer larger than x . However, as the multipath energy tends to decay as a function of the time delay, it may not be cost efficient to capture all the multipath energy from a given bit. A reduction in the filter length is therefore accomplished by setting

$$N = \left\lceil \psi \frac{T_b + (L-1)T_{\text{ch}}}{T_s} \right\rceil, \quad (8)$$

where $0 < \psi \leq 1$ is the filter length reduction compared to the filter that spans the entire multipath energy of a given bit.

The transmitted bits are estimated by hard decision on the output of the filter as

$$\hat{b}^{(1)}(j) = \text{sgn}(\mathbf{w}(j)^T \mathbf{r}(j)) \quad (9)$$

with $\mathbf{w}(j)$ being the column vector holding the filter coefficients.

In order to evaluate the performance of the LMMSE receiver with perfect knowledge about the channel and user parameters, the contribution from an unmodulated bit can seen to be

$$v^{(k)}(t) = \sum_{l=0}^{L-1} a_l^{(k)} \varphi^{(k)}(t - lT_{\text{ch}} - \tau^{(k)}) \quad (10)$$

and sampling this signal produces the vector

$$\begin{aligned} \mathbf{v}^{(k)}(m) &= [v^{(k)}(mT_b), v^{(k)}(mT_b + T_s), \dots, v^{(k)}(mT_b + (N-1)T_s)]^T. \end{aligned} \quad (11)$$

Although the expression of (4) includes all bits transmitted, only a finite number of bits, L_1 bits before and L_2 bits after

the current bit, will contribute energy to $\mathbf{r}(j)$. It is therefore possible to express $\mathbf{r}(j)$ using only the relevant bits as

$$\mathbf{r}(j) = \sum_{k=1}^K \sum_{m=-L_1}^{L_2} b^{(k)}(j+m) \mathbf{v}^{(k)}(m) + \mathbf{n}(j) \quad (12)$$

with $\mathbf{n}(j)$ holding the noise samples. The maximum bit offset that contribute energy to $\mathbf{r}(j)$ is therefore

$$L_1 = \left\lceil \frac{(L-1)T_{\text{ch}}}{T_b} \right\rceil \quad (13)$$

as the number of bits in the past influencing the decision is independent of ψ . On the other hand, the number of bits after the current bit influencing the decision is

$$L_2 = \left\lceil \psi \frac{(L-1)T_{\text{ch}}}{T_b} \right\rceil. \quad (14)$$

The LMMSE filter coefficients \mathbf{w}_o is given by the Wiener-Hopf solution

$$\mathbf{R}\mathbf{w}_o = \mathbf{p} \iff \mathbf{w}_o = \mathbf{R}^{-1}\mathbf{p}, \quad (15)$$

where \mathbf{R} is the covariance matrix and \mathbf{p} the cross-correlation vector defined as

$$\begin{aligned} \mathbf{R} &= E[\mathbf{r}(j)\mathbf{r}(j)^T], \\ \mathbf{p} &= E[b^{(1)}(j)\mathbf{r}(j)]. \end{aligned} \quad (16)$$

Applying the expectations of (16) to (12), the covariance matrix can be found to be

$$\mathbf{R} = \sum_{k=1}^K \sum_{m=-L_1}^{L_2} \mathbf{v}^{(k)}(m)\mathbf{v}^{(k)}(m)^T + \sigma^2 \mathbf{I} \quad (17)$$

with \mathbf{I} being the identity matrix. In a similar way, the cross-correlation vector is found to be

$$\mathbf{p} = \mathbf{v}^{(1)}(0). \quad (18)$$

The output of the filter is

$$\mathbf{w}_o^T \mathbf{r}(j) = \overbrace{\mathbf{w}_o^T \mathbf{v}^{(1)}(0)}^{\text{Desired}} + \overbrace{e_{\text{ISI}}(j)}^{\text{Interference}} + \overbrace{e_{\text{MAI}}(j)}^{\text{Noise}} + \overbrace{e_n(j)}^{\text{Noise}}, \quad (19)$$

where $e_{\text{ISI}}(j)$, $e_{\text{MAI}}(j)$, and $e_n(j)$ are the contributions at the output from intersymbol interference (ISI), multiple-access interference (MAI), and noise, respectively. Both $e_{\text{ISI}}(j)$ and $e_{\text{MAI}}(j)$ are approximately Gaussian as shown in [11] and $e_n(j)$ is Gaussian as the filter is linear. The BER of the LMMSE receiver may therefore be approximated by

$$\text{BER}_{\text{LMMSE}} = \frac{1}{2} \text{erfc} \left(\sqrt{\frac{|\mathbf{w}_o^T \mathbf{v}^{(1)}(0)|^2}{2(\sigma_{\text{ISI}}^2 + \sigma_{\text{MAI}}^2 + \sigma^2)}} \right) \quad (20)$$

with σ^2 being the noise variance and

$$\begin{aligned} \sigma_{\text{ISI}}^2 &= \sum_{m \neq 0} |\mathbf{w}_o^T \mathbf{v}^{(1)}(m)|^2, \\ \sigma_{\text{MAI}}^2 &= \sum_{k=2}^K \sum_{m=-L_1}^{L_2} |\mathbf{w}_o^T \mathbf{v}^{(k)}(m)|^2. \end{aligned} \quad (21)$$

4. THE ADAPTIVE LMMSE RECEIVER

Instead of implementing the LMMSE receiver by performing matrix inversion, the filter coefficients can be obtained by adaptation of the filter using an appropriate training sequence. The normalized least mean square (NLMS) and RLS algorithms are presented here only to give a better understanding of the nonrecursive formulation of the RLS algorithm presented later in this section. For all algorithms, the filter coefficients are initialized to the zero vector, that is, $\mathbf{w}(0) = \mathbf{0}$.

4.1. The NLMS algorithm

The NLMS update can be written as [12]

$$\mathbf{w}(j+1) = \mathbf{w}(j) + \kappa(j)\mathbf{r}(j)e(j), \quad (22)$$

where $e(j)$ is the a posteriori error given by

$$e(j) = b^{(1)}(j) - \mathbf{w}(j)^T \mathbf{r}(j). \quad (23)$$

The variable $\kappa(j)$ controls the effective step-size and is found as

$$\kappa(j) = \frac{\mu}{a + \mathbf{r}(j)^T \mathbf{r}(j)}, \quad a \ll E[\mathbf{r}(j)^T \mathbf{r}(j)] \quad (24)$$

with μ being the step-size bound to the interval $0 < \mu < 2$ by stability. The constant a is introduced to reduce the impact of gradient noise when $\mathbf{r}(j)^T \mathbf{r}(j)$ attains a small value. The choice of the step-size parameter μ is a tradeoff between convergence speed, and thus the needed number of training bits, and the residual error resulting in an increased BER compared to the value of (20).

4.2. The RLS algorithm

The RLS update can be written as [12]

$$\mathbf{w}(j) = \mathbf{w}(j-1) + \overbrace{\Phi^{-1}(j)\mathbf{r}(j)}^{\mathbf{k}(j)} \varepsilon(j) \quad (25)$$

with $\Phi(j)$ being the sample covariance matrix defined by

$$\Phi(j) = \frac{1}{j} \sum_{i=1}^j \mathbf{r}(i)\mathbf{r}(i)^T \quad (26)$$

and

$$\varepsilon(j) = b^{(1)}(j) - \mathbf{w}(j-1)^T \mathbf{r}(j) \quad (27)$$

being the a priori error. In order to reduce the complexity of the RLS update to approximately $O(4N^2)$ per bit, the following recursion is used:

$$\mathbf{k}(j) = \frac{\Phi^{-1}(j-1)\mathbf{r}(j)}{1 + \mathbf{r}(j)^T \Phi^{-1}(j-1)\mathbf{r}(j)}, \quad (28)$$

$$\Phi^{-1}(j) = \Phi^{-1}(j-1) - \mathbf{k}(j)\mathbf{r}(j)^T \Phi^{-1}(j-1). \quad (29)$$

Initialization of the inverse covariance matrix is done as

$$\Phi^{-1}(0) = \frac{\delta}{E[\mathbf{r}(j)^T \mathbf{r}(j)]} \mathbf{I} \simeq \frac{\delta}{\mathbf{r}(0)^T \mathbf{r}(0)} \mathbf{I}, \quad (30)$$

where δ is a regularization parameter. A value of $\delta \ll 1$ will cause a high degree of regularization whereas $\delta \gg 1$ will introduce little regularization. The choice of δ is therefore a tradeoff between reducing the noise and not constraining the adaptation.

4.3. The nonrecursive least-squares algorithm

The nonrecursive least-squares (NLS) algorithm will now be derived from the RLS update. Let the vector $\gamma(j)$ be defined as

$$\gamma(j) = \Phi^{-1}(j-1)\mathbf{r}(j) \quad (31)$$

and rewrite (29) as

$$\Phi^{-1}(j) = \Phi^{-1}(j-1) - \frac{\gamma(j)\gamma(j)^T}{\delta(j)} \quad (32)$$

with the scalar $\delta(j)$ being defined as

$$\begin{aligned} \delta(j) &= 1 + \mathbf{r}(j)^T \Phi^{-1}(j-1)\mathbf{r}(j) \\ &= 1 + \mathbf{r}(j)^T \gamma(j). \end{aligned} \quad (33)$$

Using these definitions, it is possible to rewrite the RLS update as

$$\mathbf{w}(j) = \mathbf{w}(j-1) + \gamma(j) \frac{\varepsilon(j)}{\delta(j)}. \quad (34)$$

The idea is now to rewrite (31) using (32) and expand the expression all the way back to the first iteration, that is, $j = 1$ resulting in

$$\gamma(j) = \Phi^{-1}(0)\mathbf{r}(j) + \sum_{i=1}^{j-1} \frac{1}{\delta(i)} \gamma(i)\gamma(i)^T \mathbf{r}(j). \quad (35)$$

However, instead of using the usual recursive formulation of (35), having a complexity of $O(4N^2)$, the nonrecursive version as directly outlined by (35) has a complexity of $O(3(j-1)N)$ at the j th iteration. This formulation of the RLS algorithm takes advantage of the fact that at the j th iteration, the rank of the sample covariance matrix is only $j-1$, if the initialization matrix is not considered, and only $j-1$ inner products are therefore needed to get $\gamma(j)$.

The ratio $G(j)$ between the complexity of the RLS and NLS algorithms at the j th iteration is approximately

$$G(j) \simeq \frac{4N^2}{3(j-1)N} = \frac{4N}{3(j-1)} \quad (36)$$

and the NLS algorithm is therefore beneficial if convergence is reached in less than approximately $4N/3$ iterations. Further, the complexity reduction averaged over the performed iterations is $2G(N_{\text{ite}})$ with N_{ite} being the number of iterations performed as the algorithm has a lower complexity in the first iterations. Therefore, using the overall complexity as a measure, the NLS algorithm is beneficial if convergence is reached within approximately $8N/3$ iterations.

In many signal processing problems, the rank of the covariance matrix is full or close to being full, leading to slow convergence of the RLS algorithm. If this is the case, the non-recursive implementation is not preferable over the usual recursive implementation. However, when the rank is low compared to the dimension of the covariance matrix, a considerable reduction of complexity is possible as a result of the higher speed of convergence. An example of such a problem is the adaptive receiver considered in this paper.

4.4. The windowed NLS algorithm

Another interesting aspect of the nonrecursive formulation is the possibility to limit the number of summations per iteration as

$$\gamma(j) = \Phi^{-1}(0)\mathbf{r}(j) + \sum_{i=j-D}^{j-1} \frac{1}{\delta(i)} \gamma(i)\gamma(i)^T \mathbf{r}(j), \quad i > 0, \quad (37)$$

where D is the number of terms included, resulting in a complexity of $O(3DN)$ per iteration when disregarding the initialization matrix. The algorithm now performs a minimization of the squared error over a sliding rectangular window of size D , that is,

$$\arg \min_{\mathbf{w}(j)} \left(\sum_{i=j-D-1}^j |\varepsilon(i)|^2 \right), \quad i > 0. \quad (38)$$

The algorithm is therefore termed the windowed NLS (WNLS) algorithm. Window functions other than the rectangular one specified here can of course also be used if desired. The algorithm can be considered a kind of a generalization of the NLMS and RLS algorithms as $D = 0$ equals the NLMS algorithm and $D = j-1$ equals the RLS algorithm. Values of D in between these two extremes provide algorithms with convergence speed scaling with D as the algorithm estimates the sample covariance matrix over the window. It should also be noticed that when $j \leq D+1$, the WNLS algorithm is equivalent to the NLS algorithm.

5. SYNCHRONIZATION OF THE ADAPTIVE LMMSE RECEIVER

The task of synchronizing the receiver with the transmitter and staying synchronized over time is an often-overlooked

topic compared to modulation and demodulation. However, as this is absolutely crucial to the performance of the system, a method of synchronizing the adaptive LMMSE receiver is presented here based on the same principles as used in [3].

The type of synchronization considered is the initial synchronization including both bit and frame synchronization over the UWB multipath channel in [9]. However, the problem of tracking changes between the transmitter and the receiver is not considered. It is therefore assumed that the clocks of the receiver and transmitter are the same except for an unknown offset and that the channel is stationary.

5.1. Bit synchronization

Firstly, bit synchronization can be established by taking advantage of the adaptive nature of the receiver. If at first the AWGN channel is observed, it can be noted that if the receiver is not synchronized to the transmitter, extending the filter length by one bit length can capture all energy from a desired bit. The adaptive algorithm will therefore automatically suppress coefficients outside of the correct bit interval and bit synchronization is therefore automatically achieved, but this comes at the expense of increasing the filter length to twice its original size. Increasing the filter length by a bit length in the UWB multipath channel will, in a similar way as in the AWGN channel, ensure that at least the same energy is captured as if the systems were synchronous. It is then possible to estimate the timing offset between the transmitter and receiver by observing the converged filter coefficients and use this to correct the timing in the receiver [7]. In this manner, the receiver will be able to take full advantage of the increased filter length to capture a larger part of the multipath energy, but this correction is not included in this paper.

The increase in filter length may be modeled by a larger value of ψ given by

$$\psi' = \psi + \psi_b, \quad (39)$$

where ψ determines the filter length of the fully synchronous system and ψ_b represents the increase needed to accommodate a full bit length and is given by

$$\psi_b = \frac{T_{\text{mono}} N_c}{T_s N_{\text{full}}} = \frac{N_c}{N_c + (L - 1) T_{\text{ch}} / T_{\text{mono}}}. \quad (40)$$

The AWGN channel therefore requires $\psi_b = 1$ as argued earlier and in the case of the UWB multipath channel, the value of ψ_b will typically be much less than unity and the increase in complexity will therefore be small. This is a direct consequence of the fact that the energy spread in the UWB channel is typically much larger than the bit period.

5.2. Frame synchronization

In order for the receiver to lock onto the transmitted information, the bits are arranged into a frame consisting of N_f bits. In the beginning of the frame, a known length N_t

maximal-length sequence is inserted acting as a synchronization burst to make the adaptation possible. The remaining $N_d = N_f - N_t$ bits of the frame are the information bits. However, as the receiver has no knowledge of when to look for the synchronization sequence, this ambiguity can be modeled by placing the start of the synchronization burst at a position N_s unknown to the receiver.

To acquire correct synchronization, the receiver will now have to estimate N_s . This is done by searching all possible positions of the synchronization burst and select the estimate \hat{N}_s that leads to the smallest mean square error (MSE) averaged over the performed iterations, that is,

$$\arg \min_{\hat{N}_s} \sum_{j=1}^{N_t} |b^{(1)}(j) - \mathbf{w}(j-1)^T \mathbf{r}(j + \hat{N}_s)|^2. \quad (41)$$

The receiver now uses the converged coefficients at the estimated position to detect the transmitted bits. Since the current bit influences the observation window as long as $-L_2 \leq e_s \leq L_1$, it is not required that the synchronization error $e_s = N_s - \hat{N}_s$ be zero in order to correctly detect a bit. Still, having $e_s = 0$ maximizes the received energy and thus makes it desirable to minimize $|e_s|$.

6. SIMULATION AND DISCUSSION

A number of simulations have been performed to assess the performance of the described UWB receiver in the multipath channel specified in [9].

The used monocycle is the 7th derivative of a Gaussian pulse with a pulse width $T_{\text{mono}} = 0.67$ nanosecond, as the spectrum of this pulse propagating in free space is a good match for the FCC regulations [1] giving a bandwidth in the order of 3 GHz [13]. The number of samples per monocycle was set to 13 yielding $T_s = 51.3$ picoseconds in order to provide good rejection of aliasing at half the sample rate. It may however be possible to reduce this high sampling rate by taking advantage of the aliasing in the form of sub-Nyquist sampling [8].

The system simulated consists of K sample-asynchronous users each using a length $N_c = 15$ large-set Kasami spreading code, making it possible for up to approximately 15 users to simultaneously use the system. The users do not need to have knowledge about the spreading codes used in the system, as the receiver requires only the training sequence to adapt. All users are assumed received at the same power level.

The channel model employs a tap spacing of $T_{\text{ch}} = 2$ nanoseconds with the total number of taps being $L = 100$ [9]. This results in the number of filter coefficients being $N_{\text{full}} = 4056$ if the entire energy spread in the channel is to be covered. The channel impulse response is fixed during adaptation and BER measurements, but to help average out the stochastic nature of the channel model, simulations are averaged over 10 different channels. The reason for using only 10 different channels is that it is computationally

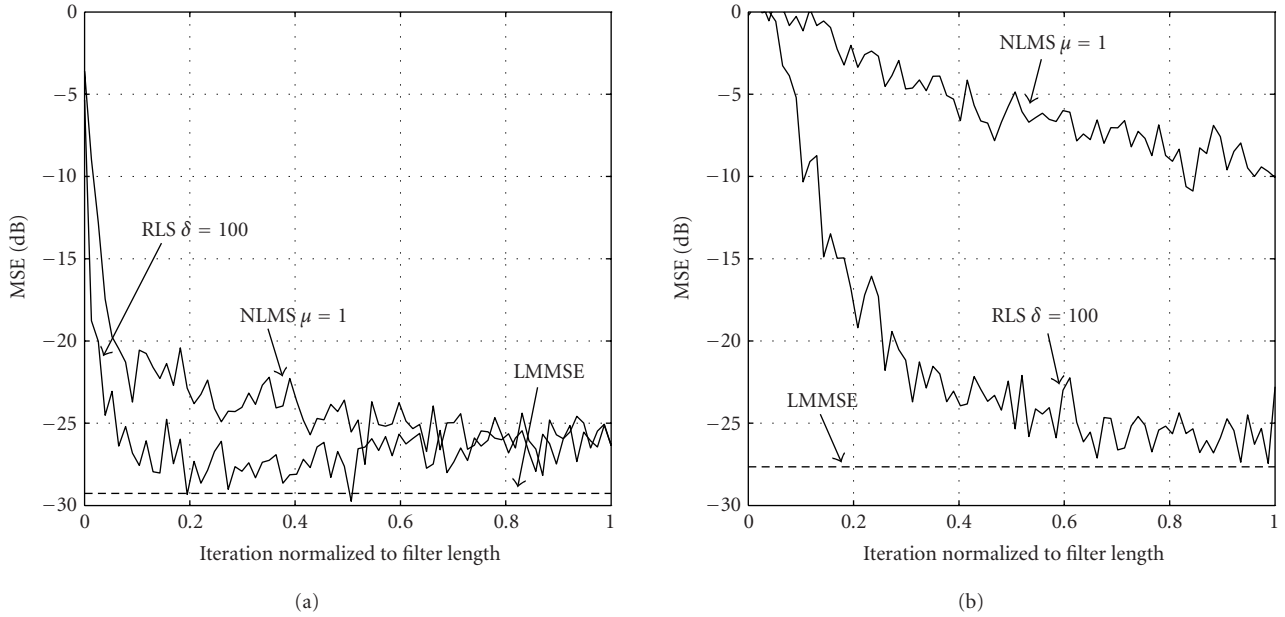


FIGURE 1: Convergence of the receiver ($N_c = 15$, $\psi = 1$, SNR = 20 dB). (a) $K = 1$ and (b) $K = 15$.

intractable to average out the entire channel and that this number of channels drawn from the model produces results being within ± 0.5 dB of the results obtained by performing the much larger number of simulations needed to average out the channel distribution.

For NLMS, a step-size of $\mu = 1$ was selected, as a smaller step-size will produce unacceptable slow convergence. In the case of RLS, the value $\delta = 100$ was chosen to minimize the effect of regularization as it is of higher importance not to constrain the adaptation when many users are active in the UWB multipath channel.

For a more in-depth description of the effects of these adaptation parameters on the performance of the system in both the AWGN and UWB multipath channel, the interested reader is referred to [13].

6.1. Convergence

The convergence behavior of the receiver is important in order to determine the number of training bits necessary and verify that the filter coefficients converge to the LMMSE solution.

Observing the convergence plotted in Figure 1, it should be noted how the addition of users makes the receiver converge more slowly as the dimension of the problem scales with the number of users. In the case of 15 users using the NLMS adaptation, the speed of convergence becomes very slow and does not reach convergence within the simulated iterations. The RLS algorithm manages to converge much faster as a result of its knowledge of the estimated inverse covariance matrix, but increasing the number of users also impacts it.

In Figure 2a, the convergence of the WNLS algorithm is plotted showing how the performance scales from NLMS to

RLS when increasing the window length, as its knowledge of the estimated inverse covariance matrix grows with the window length.

6.2. BER simulations

A series of Monte Carlo simulations have been performed to estimate the BER performance of the receiver under the assumption that the receiver has knowledge of the timing parameter $\tau^{(1)}$. The number of iterations performed is kept fixed at $N_{\text{ite}} = N_{\text{full}}$ and a total of 100 bit errors must occur before a BER value is accepted.

From Figure 3 it can be seen that under both light- and full-load conditions of 1 and 15 users, respectively, the RLS algorithm is capable of providing reasonably good performance even in the case of restricting the filter length to approximately $\psi = 0.2$. In the case of only a single user, the RLS algorithm comes very close to the LMMSE receiver, but it is not quite capable of reaching the bound when the load is increased to 15 users. The NLMS algorithm has been left out, as its general performance is unsatisfying [13], which is also clear from the slow convergence depicted in Figure 1.

6.3. Synchronization

By inserting the needed parameters in (40), the filter length can be seen to increase by $\psi_b = 0.048$ in order to let the filter span an extra bit length. Focusing on the case of $\psi = 0.2$ this results in $\psi' = 0.248$ leading to $L_1 = 20$ and $L_2 = 5$. The BER performance of the receiver with this extended filter length is plotted in Figure 3 under the assumption of being synchronized with the desired user.

The performance of the joint synchronization and detection is shown in Figure 4 assuming $N_d = 500$. Further,

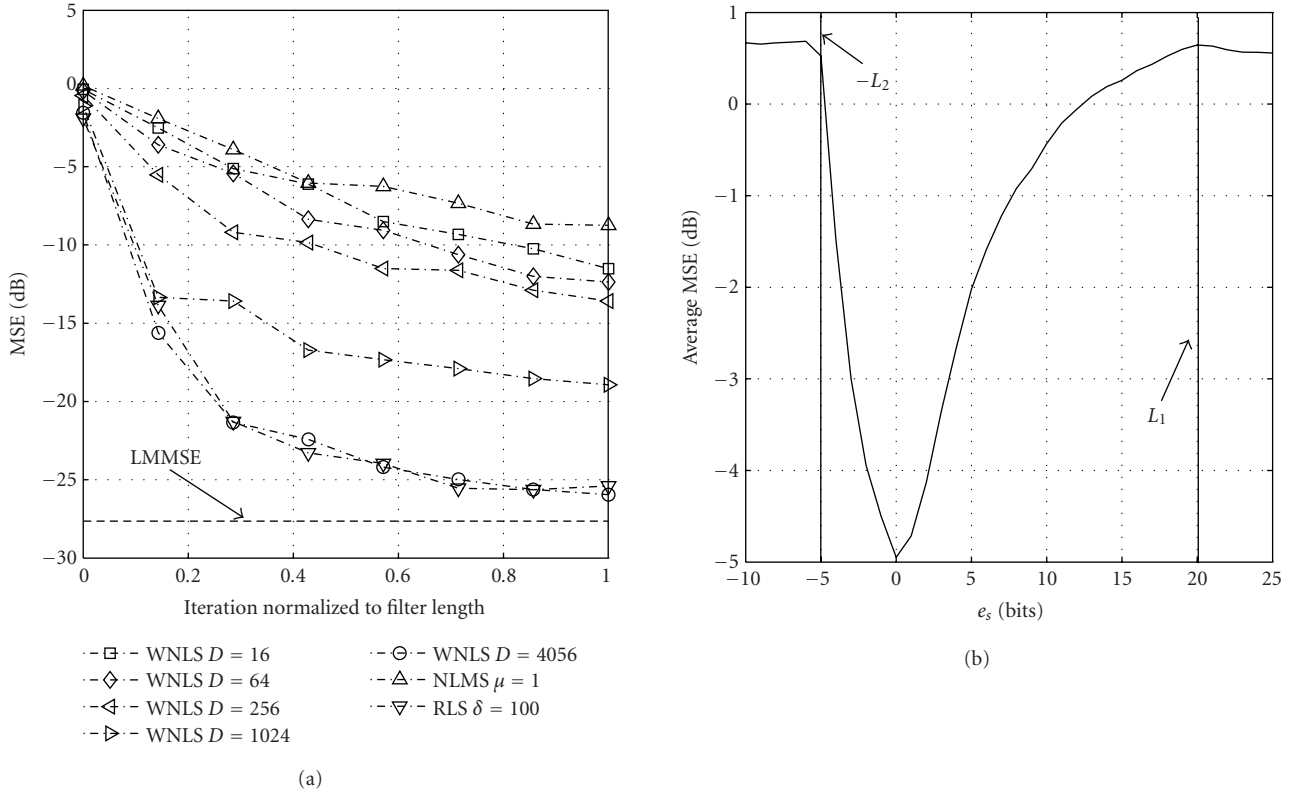


FIGURE 2: Convergence of the WNLS algorithm and the average MSE as a function of synchronization error ($N_c = 15$, $\psi' = 0.248$, $\delta = 100$). (a) $K = 15$, SNR = 20 dB and (b) $K = 1$, SNR = 10 dB, $N_{\text{ite}} = N_t = 127$.

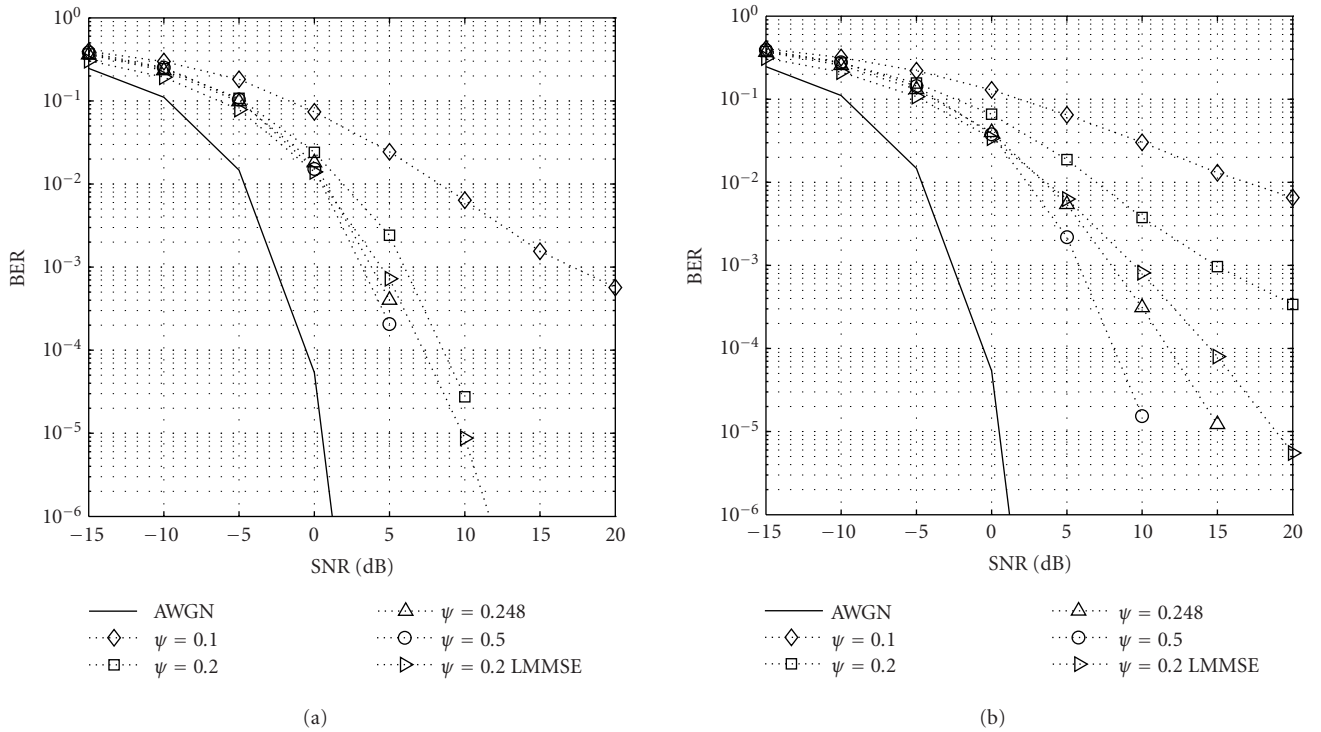


FIGURE 3: The BER in the UWB multipath channel when the receiver is synchronized to the desired user ($N_c = 15$, $N_{\text{ite}} = N_{\text{full}} = 4056$, RLS $\delta = 100$). (a) $K = 1$ and (b) $K = 15$.

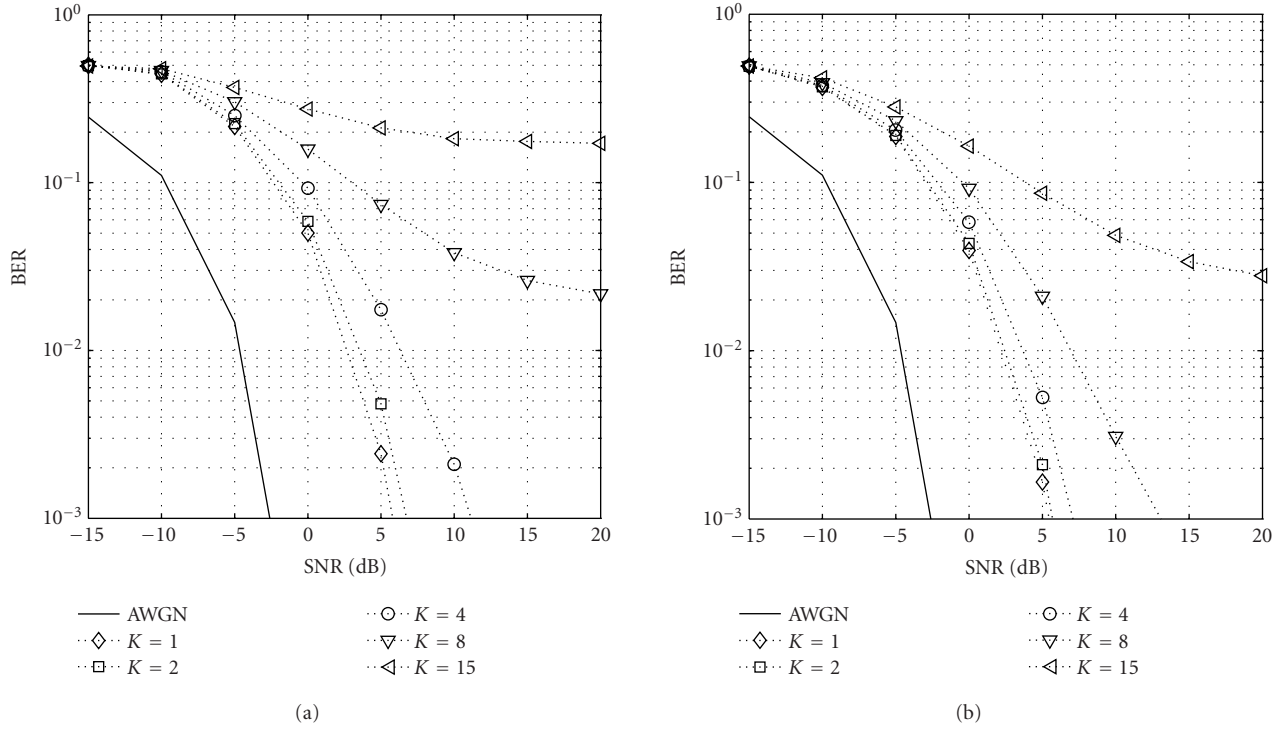


FIGURE 4: Performance of the presented joint synchronization and detection scheme using the NLS algorithm ($N_c = 15$, $\psi' = 0.248$, $\delta = 100$). (a) $N_{ite} = N_t = 127$ and (b) $N_{ite} = N_t = 255$.

Figure 2b plots the average MSE as a function of the synchronization error showing how on average the synchronization error is minimized by (41). However, the synchronization error may be nonzero and the performance of the receiver therefore degrades, as the captured energy becomes less. This, along with the fact that in the two cases shown only $N_{ite} = 127$ and $N_{ite} = 255$ iterations are performed, explains why the BER in Figure 4 degrades compared to that of Figure 3, especially when more users are added. This performance degradation is the price paid by using this low-complexity type of joint synchronization and detection. However, the achieved performance is the same as could be reached by using the RLS algorithm, but in the example where $N_{ite} = 127$, the NLS algorithm lowers the complexity by a factor of $G(N_{ite}) \approx 10$ resulting in approximately 20 times the overall complexity reduction.

7. CONCLUSION

A method for performing joint synchronization, channel estimation, and multiuser detection for single-band DS-CDMA UWB communications has been presented based on the principles in [3, 8]. Simulations of the receiver show good results in the UWB multipath channel in [9] using RLS adaptation, but the complexity of the RLS adaptation is very high. To help alleviate this problem, a novel algorithm termed the WNLS algorithm is derived, potentially lowering the computational complexity while preserving the performance of the RLS algorithm.

ACKNOWLEDGMENTS

The author would like to thank the anonymous reviewers for pointing out that the synchronization scheme was already in existence, as the author was unaware of this fact. Further, the author wishes to thank Thomas Fabricius, Spectronic Denmark, and Associate Professor Jan Larsen, Technical University of Denmark, for their various fruitful discussions. In addition, the author greatly appreciates the careful proof-reading by Pedro Højén-Sørensen, Nokia Denmark, and Ole Nørklit, Nokia Denmark.

REFERENCES

- [1] Federal Communications Commission (FCC), "Revision of part 15 of the commission's rules regarding ultra-wideband transmission systems," First Report and Order, ET Docket 98-153, FCC 02-48; Adopted: February 2002; Released: April 2002.
- [2] A. Muqaibel, A. Safaai-Jazi, B. Woerner, and S. Riad, "UWB channel impulse response characterization using deconvolution techniques," in *Proc. 45th IEEE Midwest Symposium on Circuits and Systems (MWSCAS '02)*, vol. 3, pp. 605–608, Tulsa, Okla, USA, August 2002.
- [3] U. Madhow, "Adaptive interference suppression for joint acquisition and demodulation of direct-sequence CDMA signals," in *Proc. IEEE Military Communications Conference (MILCOM '95)*, vol. 3, pp. 1200–1204, San Diego, Calif, USA, November 1995.
- [4] U. Madhow, "MMSE interference suppression for timing acquisition and demodulation in direct-sequence CDMA systems," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1065–1075, 1998.

- [5] R. Smith and S. Miller, "Acquisition performance of an adaptive receiver for DS-CDMA," *IEEE Trans. Commun.*, vol. 47, no. 9, pp. 1416–1424, 1999.
- [6] M. Latva-aho, J. Lilleberg, J. Iinatti, and M. Juntti, "CDMA downlink code acquisition performance in frequency-selective fading channels," in *Proc. 9th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '98)*, vol. 3, pp. 1476–1480, Boston, Mass, USA, September 1998.
- [7] M. El-Tarhuni and A. Sheikh, "Performance analysis for an adaptive filter code-tracking technique in direct-sequence spread-spectrum systems," *IEEE Trans. Commun.*, vol. 46, no. 8, pp. 1058–1064, 1998.
- [8] Q. Li and L. A. Rusch, "Multiuser detection for DS-CDMA UWB in the home environment," *IEEE J. Select. Areas Commun.*, vol. 20, no. 9, pp. 1701–1711, 2002.
- [9] D. Cassioli, M. Z. Win, and A. F. Molisch, "The ultra-wide bandwidth indoor channel: from statistical model to simulations," *IEEE J. Select. Areas Commun.*, vol. 20, no. 6, pp. 1247–1257, 2002.
- [10] M. Z. Win and R. A. Scholtz, "Impulse radio: how it works," *IEEE Commun. Lett.*, vol. 2, no. 2, pp. 36–38, 1998.
- [11] H. V. Poor and S. Verdu, "Probability of error in MMSE multiuser detection," *IEEE Trans. Inform. Theory*, vol. 43, no. 3, pp. 858–871, 1997.
- [12] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Upper Saddle River, NJ, USA, 3rd edition, 1996.
- [13] L. P. B. Christensen, "Signal processing for ultra-wideband systems," M.S. thesis, Informatics and Mathematical Modelling, Technical University of Denmark, Lyngby, Denmark, May 2003, <http://www.imm.dtu.dk/~lc>.

Lars P. B. Christensen was born in Copenhagen, Denmark, in November 1978. He received the M.S.E.E. degree from the Technical University of Denmark in May 2003, where he is currently working towards the Ph.D. degree. His current research interests are in the areas of digital communications and statistical signal processing.

